Semantic Annotation of Heterogeneous Data Sources: Towards an Integrated Information Framework for Service Technicians

Sebastian Bader
Karlsruhe Institute of Technology
Karlsruhe 76133, Germany

Jan Oevermann
University of Bremen &
Karlsruhe University of Applied Sciences
Karlsruhe 76133, Germany

ABSTRACT

Service technicians in the domain of industrial maintenance require extensive technical knowledge and experience to complete their tasks. Some of the needed knowledge is made available as document-based technical manuals or service reports from previous deployments. Unfortunately, due to the great amount of data, service technicians spend a considerable amount of working time searching for the correct information. Another challenge is posed by the fact that valuable insights from operation reports are not yet considered due to insufficient textual quality and content-wise ambiguity.

In this work we propose a framework to annotate and integrate these heterogeneous data sources to make them available as information units with Linked Data technologies. We use machine learning to modularize and classify information from technical manuals together with ontology-based autocompletion to enrich service reports with clearly defined concepts. By combining these two approaches we can provide an unified and structured interface for both manual and automated querying. We verify our approach by measuring precision and recall of information for typical retrieval tasks for service technicians, and show that our framework can provide substantial improvements for service and maintenance processes.

KEYWORDS

Technical Documentation, Metadata Generation, Machine Learning, Linked Data, Industrial Maintenance

1 INTRODUCTION

In modern industrial manufacturing, machines and facilities are the crucial factors for competition. Increasing their overall utilization rate is one promising strategy to reach competitive advantages. Companies with higher utilization rates can produce more products to nearly the same costs and therefore outperform the market. Therefore, efficient maintenance is a key aspect for any producing company. But in comparison to its high impact, the industrial maintenance process still heavily depends on manual work. As qualified technicians are rare and expensive, the support of their field workforce is one of the main priorities for any service provider. In this context, the efficient provision of problem-specific information is a pure necessity.

Existing reference information for service technicians can be separated into two major categories. The first one contains diligently prepared manuals, processing guidelines and training materials. We will refer to this category in the following as “controlled documents” as professionally trained technical writers create and publish them.

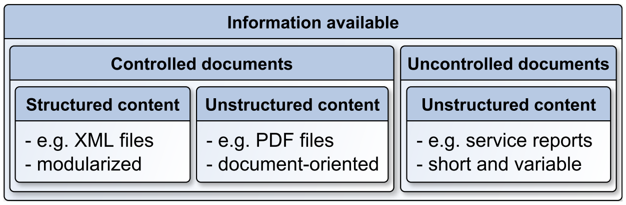

Figure 1: Available types of information

Controlled documents provide legally binding instructions and background information for facilities, components and processes. They serve as the starting point for training but also as the main source for explicit knowledge. Controlled documents can be further divided into “structured” and “unstructured” content dependent on their publication format. In technical documentation structured content is often written in semantically structured information models like DITA, DocBook or PI-Mod [9], which define self-contained content components. Typical unstructured but controlled content are document-based PDF files. Due to the generally big volume of controlled documents the targeted retrieval of required information snippets is an ongoing problem for technicians.

Other valuable information is included in machine-generated log files or manually written activity reports. We summarize these items as “uncontrolled documents” as formats may differ, content and structure depend on individual preferences and a review process is not necessarily conducted. Items of this category provide specific information on states and conducted repairs of individual machines and components. Consequently, they cover the fact that machines have an individual operation history and conducted actions by machine operators and service technicians lead to diverse settings. Yamauchi et al. categorize this collected information as “gleaning”. Uncontrolled and unstructured reports are crucial because they often contain additional information or tips, are generally more up-to-date and most importantly: “Technicians prefer gleaning to instruction following.” [32]

In this paper, we target both categories, each with an appropriate mechanism, in order to combine their respective advantages to a comprehensive view of the available knowledge. For controlled documents, we utilize methods for the classification and modularization of technical content. This allows a division into singular, coherent text modules and therefore enabling a minimal – but problem covering – presentation of information. The major challenge for uncontrolled documents is to achieve a certain degree of text quality and a minimal set of necessary meta information. As these reports are mostly already short, further separation as proposed for controlled unstructured documents is not reasonable. The main obstacle for the efficient reuse of uncontrolled documents as information artifacts is the low grammatical and syntactical quality as well as usage of of non-standardized labels and acronyms. We tackle this challenge by proposing a ontology-based automated autocompletion tool. This procedure assists the technician during the writing process and at the same time uses his direct feedback on the selected entities to annotate the report. Consequently, the reporting process is accelerated and the ratio of correctly recognized concepts increased. The higher annotation quality therefore leads to better and easier reuse of the reports as information sources.

We introduce a semantic annotation prototype for both technical documentation and short, unstructured field reports. We use semantic technologies to seamlessly combine both approaches and to allow an easy extension with additional annotation modules. We support the reuse of our system at any service provider by introducing a state-of-the-art ontology for technical categories and industrial maintenance concepts. Our system uses open knowledge graphs to increase the quality of short service reports and machine learning techniques to semantically enrich technical documentation, therefore supporting technicians at this crucial aspect of their job.

Our contribution beyond the state of the art is the combination of suitable methods for the automated or assisted annotation of content and the integration of new universal Linked Data standards into an information framework. To illustrate and validate the proposed model we use data from the sector of industrial printing presses, kindly provided from companies supporting this research. We evaluate our approach by measuring precision and recall metrics for common retrieval tasks performed by service technicians.

2 RELATED WORK

In 1994 Blumberg already stated, that an integrated data management approach can lead to a significant competitive advantage in the field of industrial maintenance [3]. By comparing more than 100 service providers, he points out how standardization and supporting systems can improve the technician's efficiency. Some of the suggested improvements, like e.g. wireless communication, have already become widely accepted. But the overall problem of supplying service technicians with the necessary information for the right task at the desired place and time is still not solved.

Yamauchi et al. claim that informal information sources like short notes and activity reports are crucial in case a non-trivial problem is faced [32]. According to their observations, field technicians first try to find an explaining story based on own experience and advice by colleagues. If this procedure fails, they continue by searching the controlled documentation base. Consequently, both documentation sources are necessary but at different stages of the process. Schweitzer and Aurich propose a continous improvement process for maintenance organization where both controlled and uncontrolled documents are shared in the maintenance network [24]. This approach especially grant customers but also suppliers with read/write access to a shared knowledge base. Although outlining the economic necessity, the authors do not propose any solution in order to effectively connect the various systems and to guarantee a common understanding of data through the whole network.

Modularizing and classifying information artifacts with semantic annotations provides two major advantages for knowledge management systems. Namely a improved retrieval by revealing the power of semantic queries and a better interoperability cross different systems can be reached. Uren et al. give an overview of manual and automated tools for semantic annotations of documents [29]. They outline that a manual annotation process is too labour-intensive and therefore must be automated. They propose three major strategies of rule- or pattern-based systems, supervised, and unsupervised machine learning approaches. They claim that the required skills to configure the automated annotation systems and the amount of effort to create training data is not always justified with a suitable annotation quality.

3 ASSISTED REPORT CREATION

All major web browsers and search engines provide basic predictions on how a started word may continue. This allows the user to increase the writing speed and also to select meaningful terms or URLs. Major parameters are past insertions of the user himself or other users. Consequently, suggestions do not regard available filter criteria and depend on trends rather than their information value, but already constitute a helpful and well accepted feature.

Semantic Autocompletion as proposed by Hyvönen and Mäkelä adds the benefits of controlled languages like clearly defined terms from a maintained vocabulary with support during the text writing process [15]. They utilize character similarity to identify related concepts in an ontology. That allows to suggest terms by their meaning and not only by e.g. their edit distance.

Tools like Magpie [10] and DBpedia Spotlight [6] annotate texts with ontology concepts. In our use case the identified entities can be used to make documents or paragraphs queryable for semantic search engines. Nevertheless, the systems require correctly formulated input. As already outlined, this is not applicable for maintenance reports where time pressure and potentially missing writing skills lead to faulty insertions. A different approach is the 'Editor' assistant[1] included in Microsofts Office 365 which enhances the previously known grammar checks and now also suggests alternative formulations.

Controlled and uncontrolled documents in terms of information sources for service technicians differ in a number of characteristics. Whereas controlled documents like e.g. manuals, product documentation or process guidelines consist of long texts ordered in structure, uncontrolled documents mostly comprise only one or two paragraphs or even sentences. Their topic is mostly not explicitly specified. The authors of uncontrolled documents (mainly service reports are regarded here), are neither trained writers nor willing to invest an additional amount of time. In order to gain the necessary textual quality to reuse the reports in future cases, domain experts with experience on the actually used terminology, abbreviations and applied process need to interpret and transform them into useful information artifacts. Although this is carried out in some cases to document knowledge of outstanding importance, a manual integration process can't cope with the high volume and heterogeneity of incoming reports in a service organization.

[1] http://blogs.office.com/2017/02/28/office-365-news-in-february-new-and-improved-intelligent-services/ accessed 17.05.2017

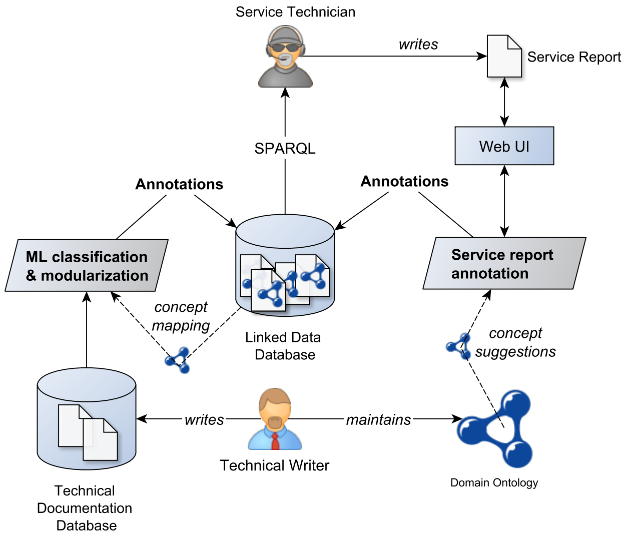

Figure 2: Model of the integrated information framework.

Two major challenges prevent current service organizations to broadly integrate service reports into their knowledge bases. First, the generally low textual and orthographic quality hampers an automated classification or identification of relevant entities. Second, the missing usage of clearly defined concepts requires a sophisticated disambiguation. This is a hard task in the mechanical and plant engineering sector, where similar terms can have totally different meanings and precise descriptions are crucial.

Based on discussions with domain experts and responsible managers of service organizations we concluded that the only practicable strategy is to support the service technician during the writing process. We propose an assisting approach where the system proposes alternative formulations. The technician selects the suggestions while continuing writing the report. By agreeing to proposed terms, the technician signals that this entity is exactly the concept he has in mind. Therefore, the precision of the documented situation is increased and at the same time the writing effort minimized. This is especially important, as the users of the system must perceive a direct, individual benefit. Consequently, we target both challenges before the reports reach the knowledge base.

3.1 Semantic Autocompletion

The user interface is divided into two parts. One area defines necessary meta information which are necessary to relate a report to a certain service order. Information on conducted actions, observed situations and informal best practices are inserted in a free text area. These statements are of high interest for service organizations as they describe which problems actually occurred, which strategies were applied and how the issue could be fixed. Other service technician in similar situations can benefit greatly from such hints.

The UI sends the last written characters whenever an input change appears. The backend transforms the incoming snippet into an autocomplete query and a fuzzy query and uses a predefined Lucene[2] index of the domain ontology to find matching entities. The search results are converted to RDF and forwarded to the UI. The technician when may or may not select a suggestion suitable to his intentions. A selection is interpreted in the way that the proposed entity is relevant to the report and was somehow involved in the conducted operation. Therefore, the entity itself, its class and assembly group are stored as matching annotations.

The entities for the suggestion process are modeled in the form of a domain ontology. Relevant concepts for service technicians are collected in the context of the STEP project and are publicly available[3]. The extracted ontology information in the form of N-Quads are the source for the Lucene index, creating a new Lucene document for each subject entity. Stored are all available literals, especially `rdfs:label`, together with classes and the entity's URI. Thereby, a match at any textual information of the entity returns all available information as included in the ontology graph. An alternatively necessary look up is therefore prevented and the response time increased.

Queried are all entities which at some point are similar to the inserted term fragment. First, an autocompletion query searches for patterns like “.*search_pattern.*”. The regarded domain is dominated by terms consisting of several words (e.g. “print ink delivery system” or “rubber blanket cylinder”) where a subset of words (e.g. “delivery system”) is not sufficiently describing components. This query type takes care to suggest more specific terms. The second query, a Lucene fuzzy query, returns similarly written entities and basically covers insertion errors and misspellings. The result sets of both queries are combined and ranked according to a matching score and visualized in a popover menu in the UI editor.

In the near future, we plan to use the recognized concepts to create a pattern of mentioned classes. Similar to the so called bridge patterns [11] these graphs allow the suggestion of more complex formulations. Even more, a selected bridge pattern contains information not only about the concepts themselves but also on the relations between them. Such information promises a deeper understanding of the described situation and allows an enhanced matching of information needs to available reports.

3.2 Methodology

A group of nine probands was confronted with ten situations based on real world maintenance tasks which are partly described in English and German. Each proband had 15 seconds to read the received service message and additional 20 seconds to understand the conducted action. Afterwards, they were asked to insert as many facts as the can remember into the Web UI of the semantic autocompletion module. In order to compare the suitability of our solution, the autocompletion functionality was only activated for half of tasks. To prevent biased results in case of easier or more complex descriptions or described situations, the provisioning with suggested terms also varied for each task and proband.

[2] http://lucene.apache.org/core/5_0_0

[3] http://people.aifb.kit.edu/mu2771/step/

4 ANNOTATION OF TECHNICAL DOCUMENTATION

The technical documentation (TD) of machinery is the most reliable and legally binding information source for service technicians. It can consist of several documents, schematics or web pages, with printed manuals as the oldest and most common form [9]. In this work we also refer to TD as “controlled documents” in the context of integrating heterogeneous data sources. Especially in the industrial sector, TD is written mostly by trained Technical Editors [28], which create content in a modularized and semantically structured manner. These self-contained information modules, called content components or topics, maximize referenced reuse across documents and decrease translation cost for globalized companies [26]. They can be published as printable documents or delivered on-demand as individual content components via online portals or smartphone apps. Although some industries legally allow the sole digital distribution of TD, printable PDF files are by far the most widespread format [28].

Classifying metadata or semantic annotations can be used to integrate content components in more dynamic scenarios, where information is aggregated or filtered automatically and utilized in facetted search [33]. Standardized classification frameworks, such as PI classification, assign taxonomic classes which can be used to identify content via semantic properties [9]. Contrary to this, legacy manuals for older machinery or new documentation from smaller manufacturers are often document-based, stored as archival formats (such as PDF) and not semantically annotated, which prevents them from targeted retrieval.

Especially in mechanical and plant engineering, the total amount of information for one product can range from several hundred to thousands of printed pages of TD alone.[4] This poses a major problem for service technicians because the retrieval of relevant information gets harder with more data. The widespread document-based presentation makes it even more difficult to efficiently search and filter the available content.

4.1 Automated classification

To annotate structured content we use an approach for the automated multi-class classification of content components used in technical communication firstly introduced in [22]. Differences in form and substance of this content type (with text size as the most notable one) require domain-specific adjustments to widespread classification techniques. The method is based on the vector space model and builds prototypical class vectors which can then be classified via cosine similarity measures.

4.1.1 Methodology. We use controlled and structured content consisting of 3686 manually classified content components in German language with an average size of 87 words as training data. In a preprocessing step, all plain text from components was extracted and unnecessary white-space, digits, special characters and punctuation were removed. Features were extracted as word groups (bigrams) and then weighted with the TF-ICF-CF method described in [22]. A content component for classification is represented as a vector $\vec{m} = (w_1, w_2, ...w_n)$ where n is the number of tokens chosen as features of the component. By supervised learning we built a $n \times c$ token-by-class matrix $M = \{w_{ij}\}$ for a set of distinct classes C, so that each class is represented as a prototypical vector. Class vectors consist of weights w_{ij} calculated from the specific distribution of a token i in class j across all content components in the training data. Our set-up is based on a vector space model, instead of more sophisticated methods (such as neural networks), for performance reasons. As multi-class classifier we chose simple *cosine similarity* [19] instead of support vector machines or naïve Bayes due to the high numbers of features and the heterogeneous size and distribution of classes [5]. The same set of parameters and configurations was used for all classification tasks independent from the underlying semantics.

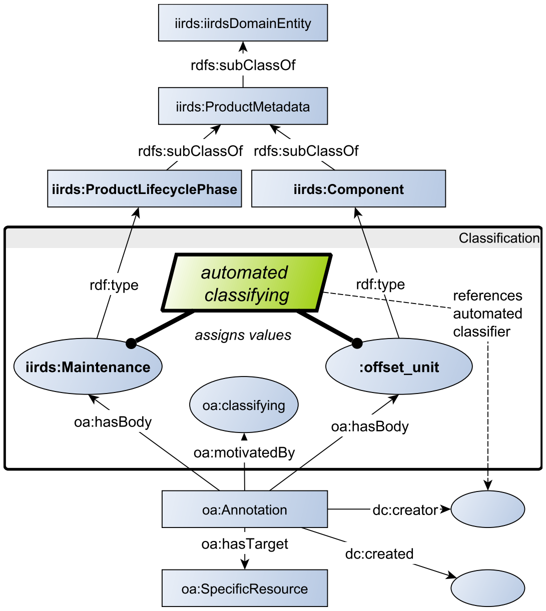

Figure 3: Annotation model of automated classifying.

4.1.2 Classification system. The domain of TD covers the writing and structuring of user manuals. Documents and sections contained therein are constrained in many ways by standards and regulatory rules. One of the most important regulations states predefined content types in the sequence of traditional chapter structures for manuals and interactive electronic technical documentation [16]. These sections are resembled by content components in CCMS. The corresponding content types follow the lifecycle of engineering products [1]. This covers, for example, information about transportation, installation and adjustment of machinery, or instructions on how to use, maintain and dispose a product (which all refer to typical lifecycle phases of a product). Additional technical data, advice on safety issues and conceptual or other descriptive information (for example about configuration, layout and functionality of the product) must also be included.

Basis for the manual classificiation is a PI classification model, which includes intrinsic classes for the type of information of a content component (e.g. “technical data” or “maintenance task”)

[4] Our example test set contains about 500 XML-based content components and about 700 pages PDF of documentation.

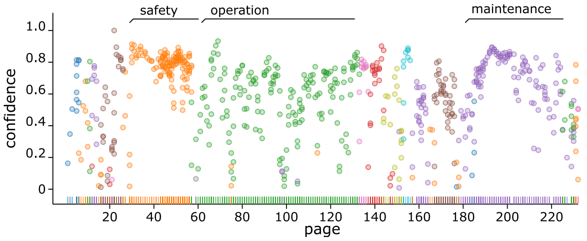

Figure 4: Example ranges derived from chunked text classification and confidence transitions (color = class).

and for which part of a product a content component is relevant (e.g. "print ink feeding" or "single sheet feeder"). Coupled with extrinsic metadata (such as the specific model of the product, for which the content is valid) it is possible to reliably filter relevant content components for specific use cases (e.g. "the maintenance of the print ink feeding for printing press TR-3B"). Technical writers assign these classes at the time of creation, usually assisted by a component content management system (CCMS).

In our test set-up we classified content with intrinsic information-related ("which type of information?") and product-related ("which part of the product?") intrinsic classes. The mapping of class values to semantic instances is described in section 5.

4.2 Automated modularization

Content which is only available in an unstructured and document-based format (such as untagged[5] PDF files) is usually excluded from granular access through filtering or faceted search. We use a method presented in [21] to reconstruct the semantic structure of a document by detecting boundaries between content of different classes. We extend the algorithm by adding a recalculation of page positions based on characters, an automated range finding (see section 4.2.2) and the transformation of results into the standardized WebAnnotation data model (see section 5). As the segmentation technique uses the same TD-specific classifier for content components we can leverage the existing setup. Because the classification solely relies on extracted text, legacy documents can be processed regardless of their visual appearance or formatting. Furthermore scanned documents without embedded text can be preprocessed with OCR techniques.

4.2.1 Methodology. Text from PDF documents was extracted with the open source library PDF.js[6] and combined into a single string while heuristically removing hyphenation and punctuation. The remaining string is then tokenized into single words by segmentation on word boundaries (spaces, line breaks).

We group the set of obtained words W in arbitrary text chunks $C = \{c_1, ..., c_n\}$, where $c_i \subset W$. The size of chunks is based on the previously collected average word count of content components in the training data, in our case $a = 87$. To distribute text chunks across the document content we chose an offset $r = 0.25$. This offset defines the starting position for each chunk. Therefore, a text chunk c_i at position i can be defined as followed (for $i > 1$): $c_i = \{W_{(i-1)*r}, W_{(i-1)*r+1}, ..., W_{(i-1)*r+a}\}$ [21].

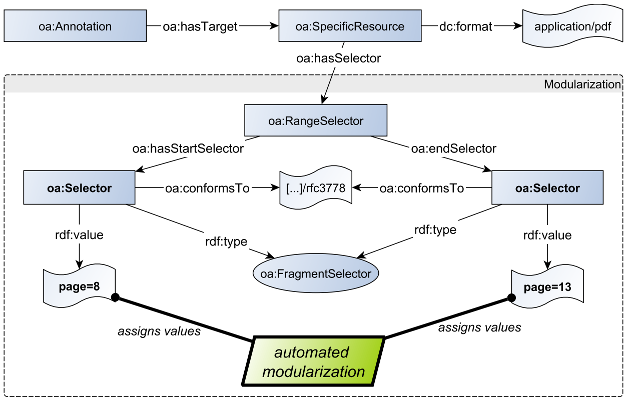

Figure 5: Annotation model of automated modularization.

Chunks are stored with additional meta information about the exact character position relative to the whole document to derive the PDF page and position a specific text appears on. After generating all text chunks, each can be classified with the same method used for content components (see section 4.1.1). In addition to the predicted class the classifier confidence is calculated as $p = \frac{s_1 - s_2}{s_1 - s_n}$ $(0 < p < 1)$ where per-class classification scores s_c for n classes c are sorted from high (1) to low (n) [22].

4.2.2 Range finding. After classification of the generated text chunk and plotting the results along the page sequence of the document (see Figure 4) several clusters of the same classification can be recognized. To annotate these parts of the document, page ranges are defined for contiguous chunks with the same classification. A range is therefore defined by a start and end page in the document and can be annotated by the predicted class and the contained text equivalent. The algorithm can be further refined by not taking outliers into account, which can appear inbetween chunks of the same class as single chunks of another class while also having a low confidence.

5 STANDARDIZED SEMANTIC ANNOTATION

To integrate the generated classifications of service reports, content components and document ranges into a universal information framework, we utilize several standards revolving around the Linked Data Platform principles [27]. We focused on reusing existing standards and technologies to make our data available for a wide range of applications and provide docking points for future extensions. In this work, we demonstrate the flexibility of our proposed semantic annotation framework by integrating content from three different data sources: content components, documents and service reports.

5.1 WebAnnotation

In February 2017 the W3C published its Recommendation for "Web Annotation" consisting of three parts: data model [30], vocabulary [31] and protocol. The specification is the result of years of ongoing efforts in the semantic web community to create an universal

[5] In this context *tagged* "defines a set of rules for representing text in the page content so that characters, words, and text order can be determined reliably" [2]
[6] https://mozilla.github.io/pdf.js

method for annotating arbitrary resources or parts thereof [14]. As the standard defines “classifying” as one of the possible motivations for annotating, it fits our need for a flexible semantic framework to integrate several annotated data sources. Annotations consist of two main parts: a “Body” (the annotation) and a “Target” (the annotated resource). Targets can further be refined by “Selectors” which can point to a specific part or excerpt of a resource. Annotations can have additional metadata (such as information about their origin) and are resources themselves.

5.2 iiRDS

The slow shift of industries towards digitization and the rising popularity of concepts like ”Industry 4.0” and ”Smart Factories” bring changes to the sector of technical documentation as a whole. A proper and unified annotation of content becomes even more important to exchange TD between components and monolithic documents must transform to more modular concepts to provide efficient ways of information retrieval. To face these challenges a new standard for the annotation and delivery of documentation content was developed by a industry-wide consortium.

In March 2017 the “European Association for Technical Communication” (tekom) published a First Public Working Draft of the “intelligent information Request and Delivery Standard” (iiRDS) [12]. Besides specifying a packaging format for documents, the standard introduces a data model [13] for the annotation of technical documentation: “iiRDS defines a taxonomy of information types and describes relations between information units as a basic ontology. Thus, iiRDS is the first standard that provides a comprehensive vocabulary for technical documentation.” [12]. The schema defines docking points (in form of a classes) to express the relationship between an information unit and a part of a product. We added this class (`iirds:Component`) to our existing product-part-ontology to link both and map intrinsic product classes (see section 4.1.2) to components defined in the ontology. Because the metadata model of iiRDS is partly based on the principles of PI classification, a 1:1 mapping between classes (see section 4.1.2) and semantic instances (see section 5.2) is possible. We map the top-level information-type classification of our training data to lifecycle phases defined in iiRDS (`iirds:ProductLifecycle Phase`).

5.3 Integration

We use the body of an annotation to directly refer to an instance of either a information type classification of iiRDS or a component of the product ontology. This way we leverage all existing standards by combining them for our use case into an integrated information framework. The target part of the annotation contains selectors to exactly define where an annotated information is stored. For the three data sources used in our test set-up, different selectors are used. Content components are directly referenced as `oa:SpecificResource` (one component per file) or with an `oa:XPathSelector` or `oa:FragmentSelector` as part of an collection of content components (multiple components per file). Classified ranges in PDF files utilize a tuple of `oa:FragmentSelector`[7] to define start and end pages of a `oa:RangeSelector` in an document (see figure 5). Service reports are stored as `oa:TextualBody` with the annotation, due to the tight integration with the user interface for autocompletion. The annotations of service reports are also connected with all selected domain ontology concepts, their classes and assembly groups (see figure 6).

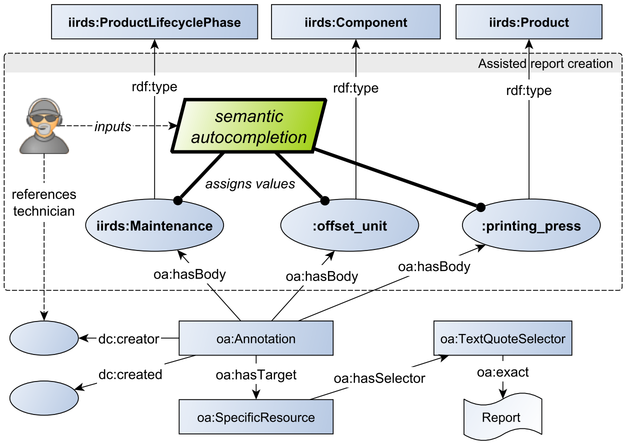

Figure 6: Annotations of service reports.

For all data sources a text excerpt (`oa:TextQuoteSelector`) is added to the target as refinement (`oa:refinedBy`) to the source-specific selectors mentioned above. This additional selector enables quick full-text search throughout all resources with native SPARQL methods and refines low resolution selectors like page ranges. Furthermore the usage of text selectors can increase robustness against modifications of the source document [4]. To comply with regulatory rules for technical documentation, a reference to the original source `oa:hasSource` is stored to allow trust-based ranking of filtering results or specific labeling of canonical sources (uncontrolled vs. controlled documents). In addition to this the origin of the annotation (`dcterms:creator`) is connected with either the software (for classifying and modularization) or the person (for autocompletion).

We store all generated annotations from service reports, content components and document ranges in a Linked Data Platform server[8] which can be queried with SPARQL. Thereby, the annotations are accessible for read, write and query operations by any client with Linked Data Platform or SPARQL support. In addition, links to external resources are provided and references to third party information resources can be added easily.

6 APPLICATION

The ongoing automation of any aspect of industrial manufacturing results in a high dependency on reliably running production facilities. But in contrast to the industry’s efforts to automate and digitize production, the maintenance process still highly depends on manual competence and experience. With the increasing variety of installed machines types and configurations the demand for the situational provision becomes a competitive advantage.

Companies already enrich their knowledge artifacts with meta data. But in order to effectively exploit the existing knowledge of an

[7] We chose fragment selectors over more granular ones because of the ubiquitous support in browsers and PDF viewers

[8] http://marmotta.apache.org/

organization, the processed and static information have to be combined with less formalized sources containing latest developments. Even more, information modules from various organizations need to be shared to make the knowledge quickly and easily accessible. Linked Data allows the simple interlinking of any type of information by Web standards and thereby access nearly anywhere and without connection barriers. The restriction to standardized annotation vocabulary guarantees the exchange of technical knowledge across companies. Especially in modern supply chains, with several component producers and high product variability, the seamless delivery of information in manageable sizes is crucial.

```
SELECT ?text WHERE {
  ?annotation oa:hasBody iirds:Maintenance> .
  ?annotation oa:hasTarget ?target .
  { ?target  oa:refinedBy ?textselector . }
  union
  { ?target oa:hasSelector  ?selector .
    ?selector oa:refinedBy  ?textselector .}
    ?textselector oa:exact ?text .
  FILTER(regex(str(?text), "keyword", "i")) }
```

Listing 1: Example SPARQL query

One of the major applications for the semantic annotation of heterogeneous data sources are Content Deliver Portals (CDP), which specialize in the targeted distribution of information for customers, internal staff and technicians [33]. Especially in the field of technical communication an increasing demand for CDPs leads to the development of competing standards for content and metadata formats. An independent and open information framework as foundation for specific clients or user interfaces can prevent a vendor lock-in and increase the exchangeability of data. In our solution clients can consume the data by either traverse the Linked Data graph or query similar to the one in listing 1.

A working prototype of the implementation of both the semantic annotator[9] and the classifier[10] is publicly available.

7 EVALUATION

We evaluate our solution in a two step process. First, the sole performance of the semantic autocompletion tool and the technical documentation classifier is regarded. The second experiment evaluates the combined system to retrieve relevant documents from both controlled documents (manuals) and uncontrolled sources (service reports).

We analyzed the inserted texts and metadata in terms of correct mentions of involved technicians and customers, affected machines and components, found error codes and conducted actions. Without the supporting autocompletion service, in average 3.13 concepts are named correctly. In contrast to that, 3.92 concepts are correctly mentioned in reports with the activated autocompletion. Even when relaxing the evaluation criteria, 3.90 concepts are similar in the sense of wrong order of terms or missing parts (e.g. only "670" instead of "ICS 670") whereas the average of nearly correct concepts with autocompletion functionality is about 4.41. Regarding the fact that each report only contains one to two sentences, the supplied functionality helps especially untrained users to better find the correct terms.

Automated classification and modularization and therefore the quality of the generated semantic annotations are subject to classifier performance. To measure the output quality we did a 10-fold cross validation and calculated the average accuracy [25] of classifications which resulted in 85.3% ± 2.5 for information type classes ($n = 6$) and 82.5% ± 2.1 for product-related classes ($n = 28$). The training set we used contained 3686 XML-based content components. Low accuracy in classification can negatively influence the general evaluation metrics precision (false positives wrongly classified) and recall (false negatives wrongly classified). We account for that by additionally running our test queries against manually classified content from the training data to get a baseline for relevancy.

The overall system combined of both modules is tested with ten queries based on real-world service requests from maintenance providers. We compare the recall and precision of keyword queries with an extended query enhanced with filtering for semantic annotations. The main aspects of interest are the kind of searched information (documents about maintenance, technical data, operation, etc.) and the assembly group or component.

We compare three variants for each query. First, we conduct a simple keyword query on the LDP SPARQL endpoint with the SPARQL 'FILTER' functionality and a regex search. The second query type combines both the keyword and the filter term in an additional filter statement. Obviously, these two query types work on any knowledge base and do not profit from our proposed automatic annotations of information modules. As a consequence, the achieved results can serve as a baseline for the third query type. These queries actually utilize the output of our integrated information framework. Listing 1 contains such a query where the keyword search – through the filter clause – is further restricted by the statement in the second line. Thereby, in this example only information modules with an `iirds:Maintenance` annotation are regarded.

A simple keyword query without any additional filtering returns the most information modules. Consequently, the recall value of these queries is always very close to 100%. But as many retrieved texts do not contain valuable information in regard to the query, only a low precision value can be reached. As a matter of fact the average F-measure is only around 0.46. Searching for occurrences of the keyword in combination with an additional filter criterion leads to different observations. Now, the amount of retrieved information modules is very small, as the filter criterion usually does not often occur in the texts itself. E.g. documents and reports about maintenance actions do generally not include the term 'maintenance' in every paragraph. This results to a significantly reduced amount of retrieved modules, increasing the precision but leading to a low recall. In total, a F-measure of no more than 0.42 is achieved.

Combining a keyword query with filtering of standardized annotations yields to a higher recall than any of the baseline query types. Even though some incorrect information modules are found (decreasing the precision in comparison to the second query type), this is compensated by the recall gain. With an average precision of 0.96 and an average recall of 0.74, the resulting F-measure of 0.80 clearly beats the baseline queries. This result supports our

[9] https://github.com/sebbader/SemanticAutocompletion
[10] https://github.com/j-oe/semantics-demo
hosted demo: http://semantics.fastclass.de

assumption that an information framework like the one proposed here can – even though it is still in a prototypical stage – make a significant difference in the field. All queries and evaluation measures are published in a GitHub repository.[11] As the query results contain confidential data they can not be made accessible.

8 CONCLUSION

By leveraging Linked Data principles and existing vocabularies we developed a standardized model for the semantic annotation of digital technical documentation and service reports. We successfully combined two approaches to tackle these challenges: For unstructured reports we use an ontology-based autocompletion for the assisted input of service information; for controlled content we utilize machine learning methods to make existing technical documentation semantically accessible through automated classification of content components and modularization of document-based formats. Both approaches are combined with state-of-the-art Linked Data standards into an integrated information framework. A first evaluation of our proposed model shows that standardized semantic annotations can improve handling of heterogeneous types of data and can therefore reduce operating time of service technicians.

In future work we want to validate the flexibility of the proposed model by adding more external data sources. In order to exploit the benefits of Linked Data we want to enrich the annotation objects with knowledge provided by open sources as the Linked Open Data Cloud and supply a fast interlinking model for closed-source corporate ontologies. Furthermore we want to focus on more complex queries which combine several annotations and couple them with a powerful full-text search. Also, we plan to implement the proposed framework as backend for service information applications.

ACKNOWLEDGMENTS

We would like to thank Wolfgang Ziegler (Karlsruhe University of Applied Sciences), Christoph Lüth (University of Bremen) and Maria Maleshkova (KIT) for discussions and support.

The research and development project that forms the basis for this report is partly funded under project No. 01MD16015B within the scope of the Smart Services World technology program run by the Federal Ministry for Economic Affairs and Energy and is managed by the project management agency at the German Aerospace Center (DLR-PT).

REFERENCES

[1] 2006/42/EC. 2006. Machinery directive of the European Parliament and of the Council. (2006).

[2] Adobe Systems (Ed.). 2001. *PDF reference: Adobe portable document format version 1.4* (3rd ed ed.). Addison-Wesley, Boston.

[3] Donald F. Blumberg. 1994. Strategies for Improving Field Service Operations Productivity and Quality. *The Service Industries Journal* 14, 2 (1994), 262–277.

[4] AJ Brush, David Bargeron, Anoop Gupta, and Jonathan J Cadiz. 2001. Robust annotation positioning in digital documents. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 285–292.

[5] Fabrice Colas, Pavel Paclík, Joost N. Kok, and Pavel Brazdil. 2007. Does SVM Really Scale Up to Large Bag of Words Feature Spaces? In *Advances in Intelligent Data Analysis VII*, Michael R. Berthold, John Shawe-Taylor, and Nada Lavrač (Eds.). Vol. 4723. Springer Berlin Heidelberg, Berlin, Heidelberg, 296–307.

[6] Joachim Daiber, Max Jakob, Chris Hokamp, and Pablo N. Mendes. 2013. Improving Efficiency and Accuracy in Multilingual Entity Extraction. In *Proceedings of the 9th I-SEMANTICS*. ACM, New York, NY, USA, 121–124.

[7] Stephen Dill, Nadav Eiron, David Gibson, Daniel Gruhl, R Guha, Anant Jhingran, Tapas Kanungo, Sridhar Rajagopalan, Andrew Tomkins, John A Tomlin, and others. 2003. SemTag and Seeker: Bootstrapping the semantic web via automated semantic annotation. In *Proceedings of the 12th WWW*. ACM, 178–186.

[8] Anastasia Dimou, Ruben Verborgh, Miel Vander Sande, Erik Mannens, and Rik Van de Walle. 2015. Machine-interpretable dataset and service descriptions for heterogeneous data access and retrieval. In *Proceedings of the 11th International Conference on Semantic Systems SEMANTiCS2015*. ACM, 145–152.

[9] Petra Drewer and Wolfgang Ziegler. 2011. *Technische Dokumentation*. Vogel, Würzburg.

[10] Martin Dzbor, Enrico Motta, and John Domingue. 2004. Opening up magpie via semantic services. In *International Semantic Web Conference*. Springer, 635–649.

[11] Basil Ell and Andreas Harth. 2014. A language-independent method for the extraction of RDF verbalization templates. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*. 26–34.

[12] European Association for Technical Communication - tekom e.V. 2017. iiRDS RDF Schema - First Public Working Draft, 03 April 2017. (2017).

[13] European Association for Technical Communication - tekom e.V. 2017. iiRDS Specification - intelligent information Request and Delivery Standard - First Public Working Draft, 03 April 2017. (2017). https://iirds.tekom.de

[14] Bernhard Haslhofer, Robert Sanderson, Rainer Simon, and Herbert van de Sompel. 2012. Open annotations on multimedia Web resources. *Multimedia Tools and Applications* (May 2012). https://doi.org/10.1007/s11042-012-1098-9

[15] Eero Hyvönen and Eetu Mäkelä. 2006. Semantic autocompletion. In *Asian Semantic Web Conference*. Springer, 739–751.

[16] IEC 82079-1. 2012. Preparation of Instructions for Use - Structuring, Content and Presentation. (2012).

[17] Atanas Kiryakov, Borislav Popov, Ivan Terziev, Dimitar Manov, and Damyan Ognyanoff. 2004. Semantic annotation, indexing, and retrieval. *Web Semantics: Science, Services and Agents on the World Wide Web* 2, 1 (Dec. 2004), 49–79.

[18] H. López-Fernández, M. Reboiro-Jato, D. Glez-Peña, F. Aparicio, D. Gachet, M. Buenaga, and F. Fdez-Riverola. 2013. BioAnnote: A software platform for annotating biomedical documents with application in medical learning environments. *Computer Methods and Programs in Biomedicine* 111, 1 (July 2013), 139–147.

[19] Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of statistical natural language processing*. MIT Press, Cambridge, Mass.

[20] Margaret Mitchell, Kathleen McCoy, David McDonald, and Aoife Cahill. Crowdsourcing Language Generation Templates for Dialogue Systems. In *Proceedings of the INLG and SIGDIAL 2014 Joint Session* (2014). 172–180.

[21] Jan Oevermann. 2016. Reconstructing Semantic Structures in Technical Documentation with Vector Space Classification. In *Posters and Demos Track of the 12th International Conference on Semantic Systems*, Vol. 1695. CEUR-WS, Germany.

[22] Jan Oevermann and Wolfgang Ziegler. 2016. Automated Intrinsic Text Classification for Component Content Management Applications in Technical Communication. In *Proceedings of the 2016 ACM Symposium on Document Engineering*. ACM Press, Vienna, Austria, 95–98.

[23] Laura Perez-Beltrachini, Rania Mohamed Sayed, and Claire Gardent. Building RDF Content for Data-to-Text Generation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers* (2016). 1493–1502.

[24] E Schweitzer and JC Aurich. 2010. Continuous improvement of industrial product-service systems. *CIRP Journal of Manufacturing Science and Technology* 3, 2 (2010), 158–164.

[25] Marina Sokolova and Guy Lapalme. 2009. A systematic analysis of performance measures for classification tasks. *Information Processing & Management* 45, 4 (2009), 427–437.

[26] Axel J. Soto, Abidalrahman Mohammad, Andrew Albert, Aminul Islam, Evangelos Milios, Michael Doyle, Rosane Minghim, and Maria Cristina Ferreira de Oliveira. 2015. Similarity-Based Support for Text Reuse in Technical Writing. In *Proceedings of the 2015 ACM Symposium on Document Engineering (DocEng '15)*. ACM, New York, NY, USA, 97–106.

[27] Steve Speicher, John Arwe, and Ashok Malhotra. 2009. Linked Data Platform 1.0. (2009). http://www.w3.org/TR/ldp/ 26 February 2015. W3C Recommendation.

[28] Daniela Straub. 2016. *Branchenkennzahlen für die Technische Dokumentation 2016 (Studie)*. tcworld GmbH, Stuttgart, Germany.

[29] V Uren, P Cimiano, J Iria, S Handschuh, M Vargasvera, E Motta, and F Ciravegna. 2006. Semantic annotation for knowledge management: Requirements and a survey of the state of the art. *Web Semantics: Science, Services and Agents on the World Wide Web* 4, 1 (Jan. 2006), 14–28.

[30] W3C. 2017. Web Annotation Data Model - W3C Recommendation 23 February 2017. (2017). https://www.w3.org/TR/2017/REC-annotation-model-20170223/

[31] W3C. 2017. Web Annotation Vocabulary - W3C Recommendation 23 February 2017. (2017). https://www.w3.org/TR/2017/REC-annotation-vocab-20170223/

[32] Yutaka Yamauchi, Jack Whalen, and Daniel G. Bobrow. 2003. Information Use of Service Technicians in Difficult Cases. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '03)*. ACM, 81–88.

[33] Wolfgang Ziegler and Heiko Beier. 2015. Content delivery portals: The future of modular content. *tcworld e-magazine* 02/2015 (2015).

[11] https://github.com/j-oe/semantics-queries